

Práctica 2: Clonar la información de un sitio web

Carlos Sánchez Martínez

24 de marzo de 2017

Índice

1. Probar el funcionamiento de la copia de archivos por ssh.	3
2. Clonado de una carpeta entre las dos máquinas.	3
3. Configuración de ssh para acceder sin que solicite contraseña.	4
4. Ejecute el monitor de “System Performance” y muestre el resultado. Incluya capturas de pantalla comentando la información que aparece.	5

Índice de figuras

1.1. Se muestra el directorio de la máquina 2 vacío.	3
1.2. Se muestra la ejecución del comando para la copia del archivo prueba desde máquina 1 hasta máquina 2.	3
1.3. Se muestra el archivo prueba copiado como prueba.tgz	3
2.1. Se ha clonado el directorio /var/www de la máquina 1 en el directorio /var/www de la máquina 2.	4
3.1. Ejecución del comando ssh <i>dirección máquina 1</i> . carlosubuntu es el terminal de la máquina 1. (En este caso tiene el mismo nombre de la máquina 2.)	4

Índice de tablas

1. Probar el funcionamiento de la copia de archivos por ssh.

Para copiar el archivo .tar de la máquina 1 a la máquina dos usaremos el comando `tar czf - directorio | ssh dirección máquina 2 'cat > /directorio.tar'`. Para ello se ha realizado las siguientes capturas de pantalla.

Primero se mostrará el directorio de la máquina 2 vacío.

```
carlos@ubuntu:~$ ls
carlos@ubuntu:~$
```

Figura 1.1: Se muestra el directorio de la máquina 2 vacío.

Se muestra la ejecución del comando antes mencionado para copiar, desde máquina 1, el archivo prueba que se mostrará en la máquina 2 como prueba.tgz.

```
carlos@ubuntu:~$ tar czf - prueba | ssh 10.0.2.4 'cat > ~/prueba.tgz'
carlos@10.0.2.4's password:
carlos@ubuntu:~$
```

Figura 1.2: Se muestra la ejecución del comando para la copia del archivo prueba desde máquina 1 hasta máquina 2.

Se mostrará el archivo copiado en el directorio de la máquina 2.

```
carlos@ubuntu:~$ ls
prueba.tgz
carlos@ubuntu:~$ _
```

Figura 1.3: Se muestra el archivo prueba copiado como prueba.tgz

2. Clonado de una carpeta entre las dos máquinas.

Para clonar un archivo desde una máquina a otra lo que se ha hecho ha sido clonar el directorio `/var/www` de la máquina 1 en la máquina 2. Para ello se ha utilizado la herramienta `rsync` y se ha utilizado el comando `rsync -avz -e ssh dirección máquina1:/var/www directorio destino`.

El resultado de ejecutar el comando ha sido el siguiente:

```
carlos@ubuntu:~$ ls -l /var/www
total 8
drwxr-xr-x 2 carlos carlos 4096 mar 21 16:36 html
drwxr-xr-x 3 carlos carlos 4096 feb 23 17:21 www
carlos@ubuntu:~$
```

Figura 2.1: Se ha clonado el directorio /var/www de la máquina 1 en el directorio /var/www de la máquina 2.

3. Configuración de ssh para acceder sin que solicite contraseña.

Para acceder desde una máquina remota a otra sin contraseña con ssh se realizarán los siguientes pasos:

1. Vamos a acceder a la máquina 1 desde la máquina 2 para ello primero ejecutaremos el comando `ssh-keygen -b 4096 -t rsa` en la máquina 2.
2. Ahora se habrá creado un archivo con una llave publica lo añadiremos al archivo `/.ssh/authorized_keys` y se le añadirán los permisos 600.
3. Luego ejecutaremos el comando `ssh-copy-id dirección máquina 1` para hacer la copia de la llave.
4. Ejecutamos el comando `ssh dirección máquina 1` para poder hacer cambios desde la máquina 2 en la máquina 1.

Tras la ejecución de estos pasos y ejecutando el último paso `ssh dirección de máquina 1` nos queda el resultado siguiente:

```
carlos@ubuntu:~$ ssh 10.0.2.5
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Pueden actualizarse 51 paquetes.
36 actualizaciones son de seguridad.

Last login: Fri Mar 24 18:53:16 2017 from 10.0.2.4
carlos@ubuntu:~$
```

Figura 3.1: Ejecución del comando `ssh dirección máquina 1`. carlos@ubuntu es el terminal de la máquina 1. (En este caso tiene el mismo nombre de la máquina 2.)

Para salir de la terminal de máquina 1 se ejecuta el comando `exit`.

4. Establecer una tarea en cron que se ejecute cada hora para mantener actualizado el contenido del directorio /var/www entre las dos máquinas.

Para crear una tarea vamos a modificar el archivo crontab y añadir la siguiente línea:

```
0 * * * * rsync -avz -e ssh 10.0.2.5:/var/www /var/www --report /etc/cron.hourly
```

```
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /e
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /e
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /e
0 * _ * * * root    rsync -avz -e ssh 10.0.2.5:/var/www /var/www --report /e
#
```

[16 líneas escritas]

Ver ayuda Guardar Buscar Cortar Texto Justificar Posición

Figura 4.1: Fichero /etc/crontab para actualizar /var/www cada hora a las en punto.